An Introduction to Computer Vision for AR

Computer Vision is the key technology that enables Augmented Reality. It gives AR apps the ability to detect and track images, objects, and 3D spaces, essential for creating believable AR experiences.

Pp

by Poornima

AR relevant Computer vision Techniques

- Marker tracking
- Muti camera infrared tracking
- Natural feature tracking by detection
- Incremental tracking
- Simultaneous localization and mapping (SLAM)
- Outdoor tracking

Marker Tracking

- Camera representation, contour based shape detection, pose estimation, non-linear refinement

QR Codes

The simplest form of marker tracking. The app detects a QR code and places the AR content over it. Very accurate, but limited to small markers.

AR Markers

A dedicated marker that triggers AR content once detected. Accurate, but requires the user to have the marker.

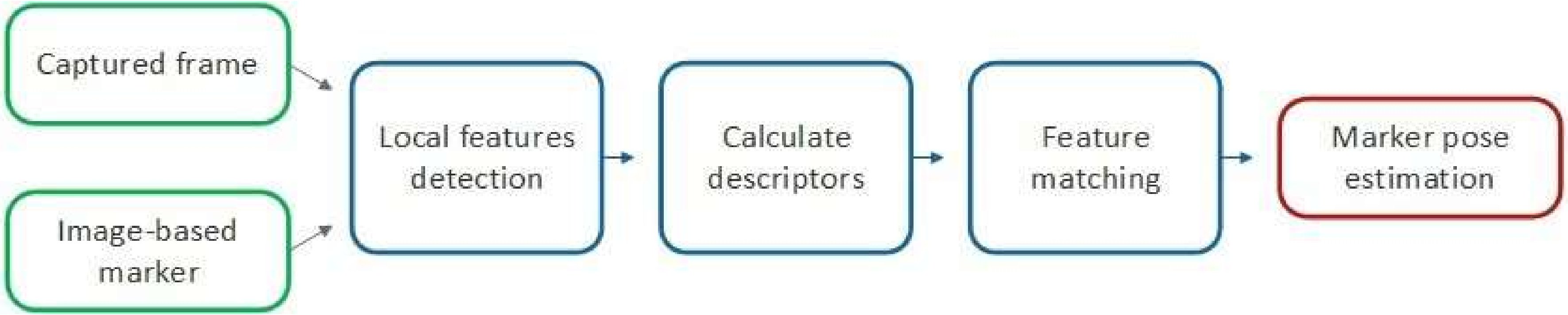
Image Recognition

The app searches the camera feed for an an image and attaches the AR content to it. to it. Very powerful, but requires a specific specific image.

- Maker-based AR works by scanning a marker which triggers an augmented experience (whether an object, text, video or animation) to appear on the device

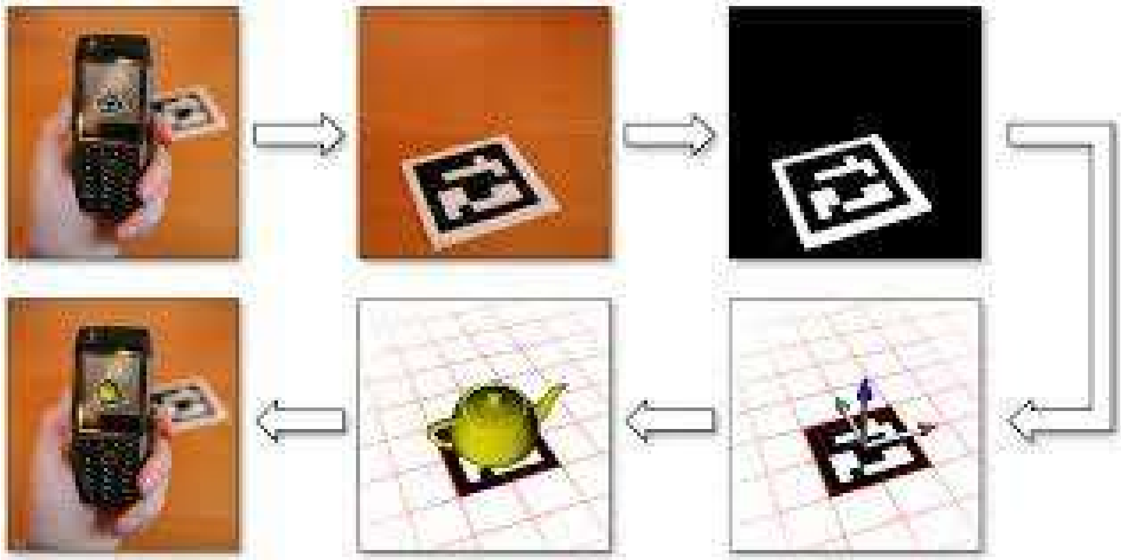
- **Markerless AR** is now the preferred image recognition method for AR applications.

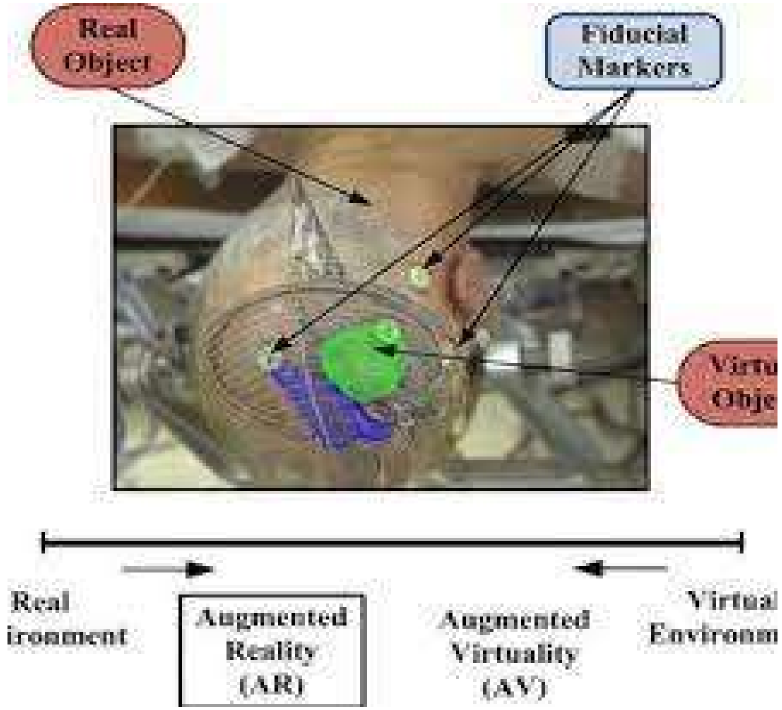
- **Marker-based AR** use markers (target images) to indicate things in a given space.

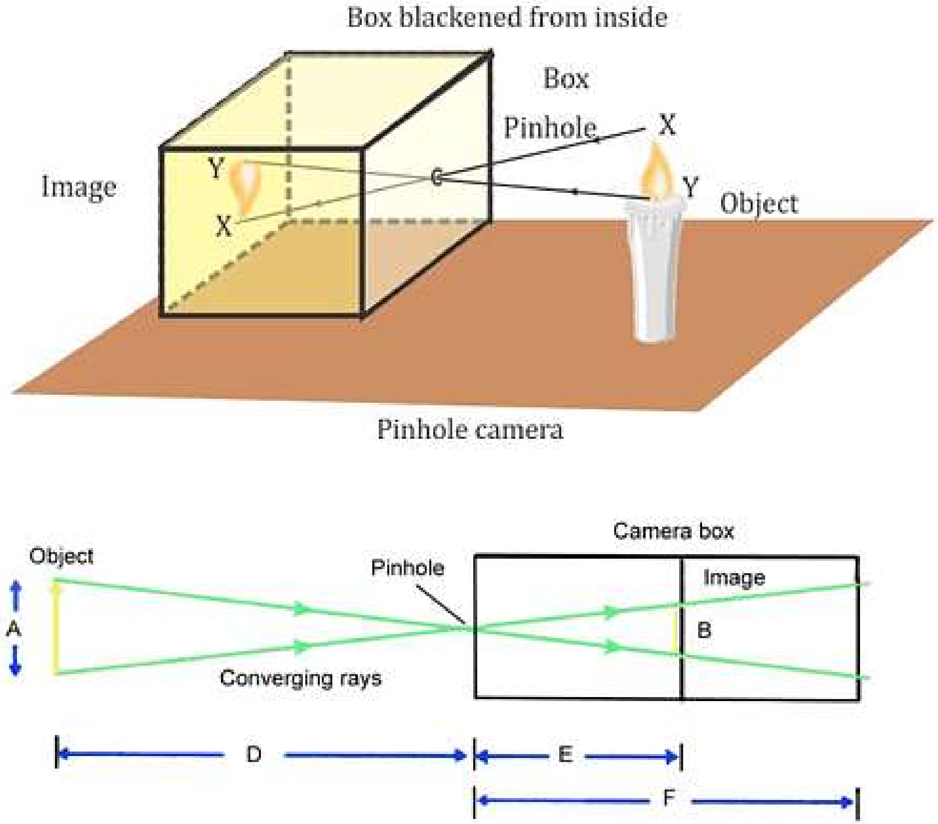


Pipeline stages of marker tracking:

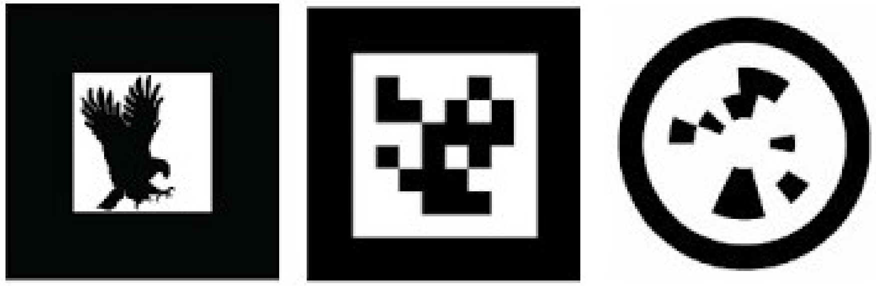
- Image capture using camera with known representation
- Marker detection by searching for quadrilateral shapes
- Pose detection from homography
- Pose refinement by non-linear reprojection error minimization
- AR rendering with the recovered pose



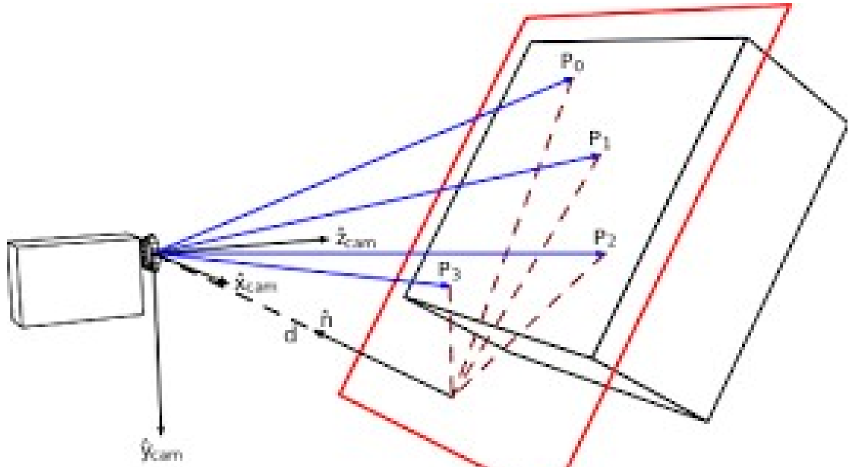




Pinhole camera model



(a) Template Marker (b) Bar-code Marker (c) Circular Marker



Homography

Multiple-Camera Infrared Tracking

- Multi view geometry knowing 2D-2D point correspondences in multiple camera images, triangulation, orientation

1

How it works

Multiple cameras placed in the AR device send infrared beams to each other. These beams help measure the distance between the cameras and other objects, essential for creating accurate AR experiences.

2

Pros

Highly accurate, provides seamless AR experience in any lighting condition. Ideal for indoor AR experiences.

3

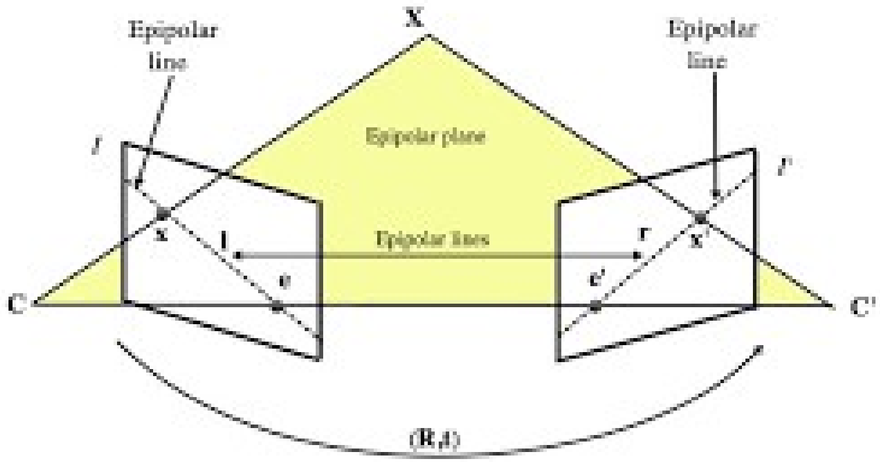
Cons

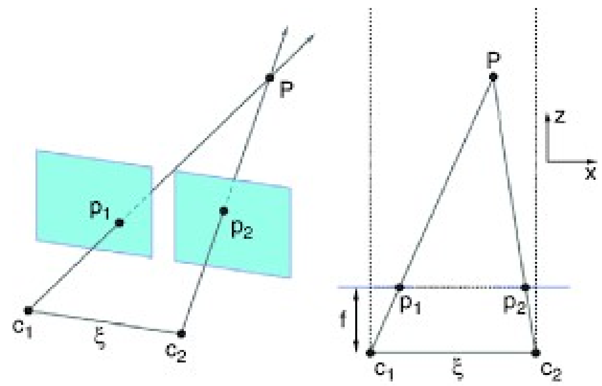
Expensive, requires specific hardware, limited to indoor use, may interfere with other IR devices.

- In general, the **known points in the world** will not be constrained to a plane, as assumed in markers.
- For **tracking arbitrary objects**, require general **pose estimation**, which addresses the problem of determining the camera pose from 2D-3D correspondences between known points q, in world coordinates and their projections p, in image coordinates.
- Simple **infrared tracking system** designed to **track rigid body markers** composed of four or more retro-reflective spheres.
- It uses **an outside-in setup** with multiple infrared cameras.
- A **minimum of two cameras** in a known configuration (a calibrated stereo camera rig) is required.
- With this strategy, the additional input and **wider coverage** of the scene from multiple viewing angles will improve the tracking quality and the working volume.
- In practice, four cameras set up in the corners of a laboratory space are a popular configuration. Use of more than two cameras will improve the performance of the system, but is not fundamentally different from the stereo case.

The stereo camera **tracking pipeline** consists of the following steps:

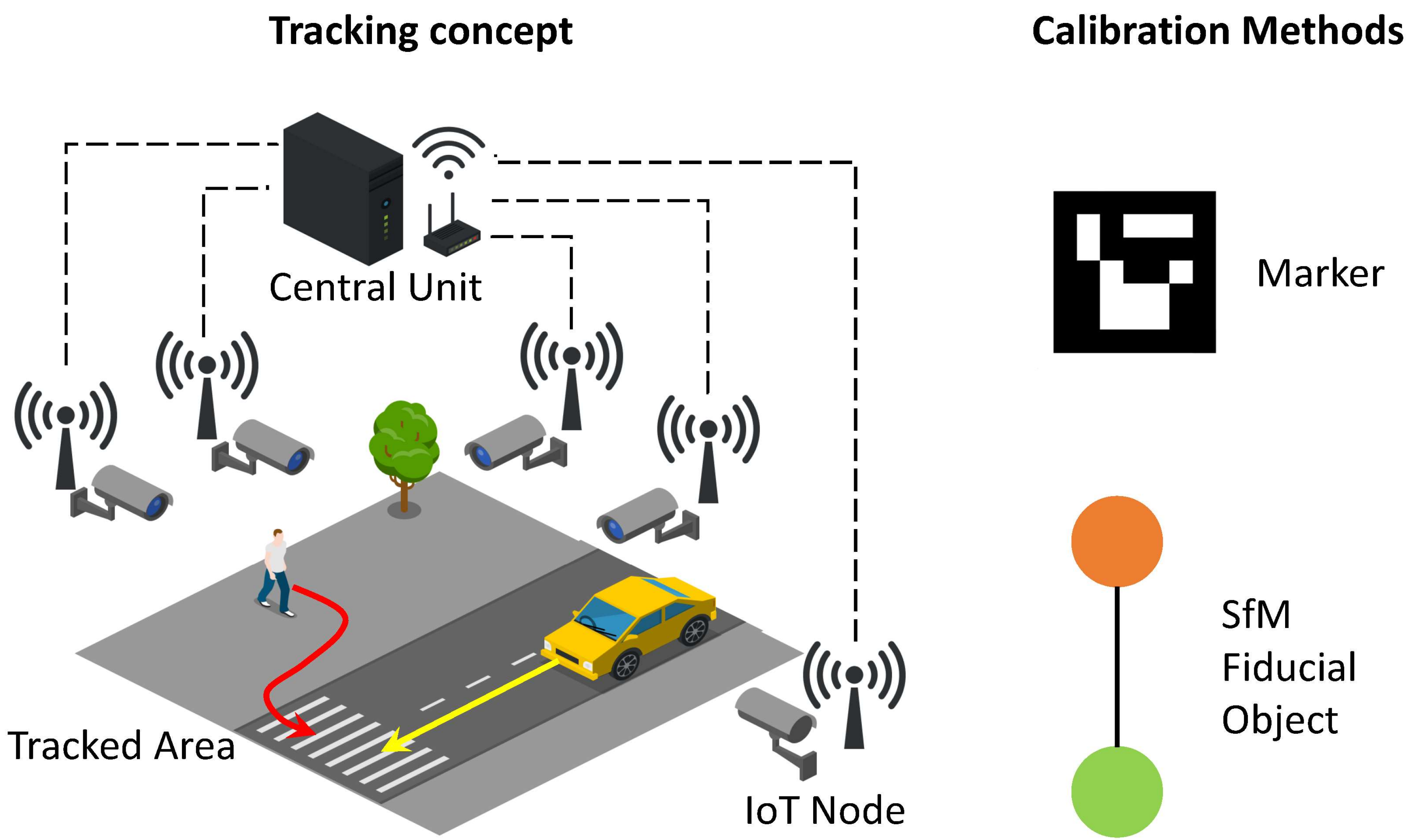
1. Blob detection in all images to locate the spheres of the rigid body markers
2. Establishment of point correspondences between blobs using **epipolar geometry** between the cameras
3. Triangulation to obtain 3D candidate points from the multiple 2D points
4. Matching of 3D candidate points to 3D target points
5. Determination of the target's pose using **absolute orientation**





Epipolar

Triangulation


Tracking concept
Calibration Methods
Central Unit
Marker
Tracked Area
IoT Node
SfM
Fiducial
Object

Natural Feature Tracking by Detection

- To recognize physical objects in markerless AR
- Point and region features are automatically and adaptively selected for properties that lead to robust tracking
- Does Point detection in images, creation & matching of descriptors, pose computation from 2D-3D correspondences
- Images must have reasonable
  - degree of detail, sharp edges, good resolution, shape

How it works

The app detects the natural features of the environment - corners, edges, and lines - and uses them as landmarks for AR content. Ideal for creating realistic outdoor AR experiences.

Pros

Uses the natural environment, accessible, low-cost, versatile, and perfect for outdoor AR experiences.

Cons

May not work in low light conditions, complex scenes, or identical environments. May require extensive processing power.

Pipeline steps:

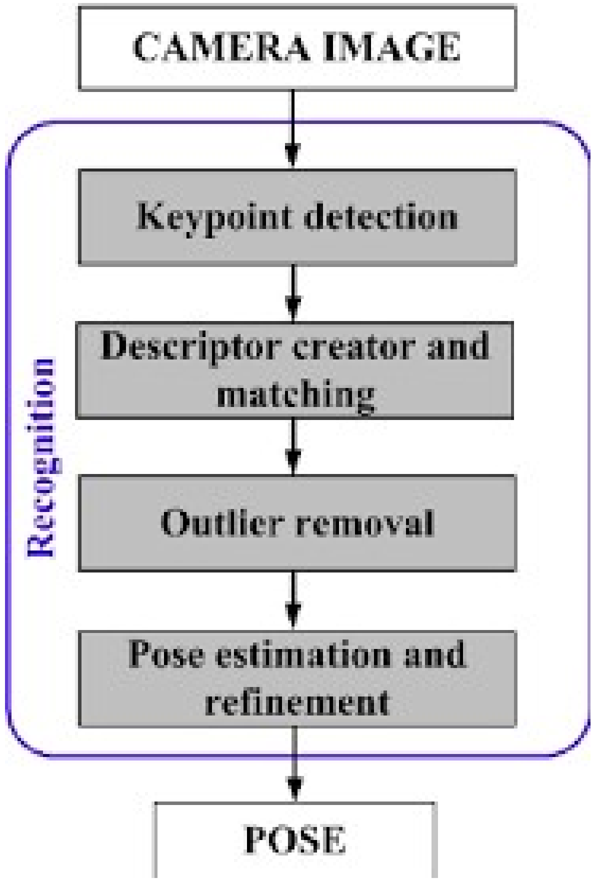
1. Interest point detection
   – good point / feature independent of any observed parameters like viewpoint & illumination
   - using Harris corners / DOG/ FAST detectors

2. Descriptor creation
   – data structure matching interest points to tracking model
   - Image patch around interest point (not invariant to rotation & scale)
   - using SIFT descriptor

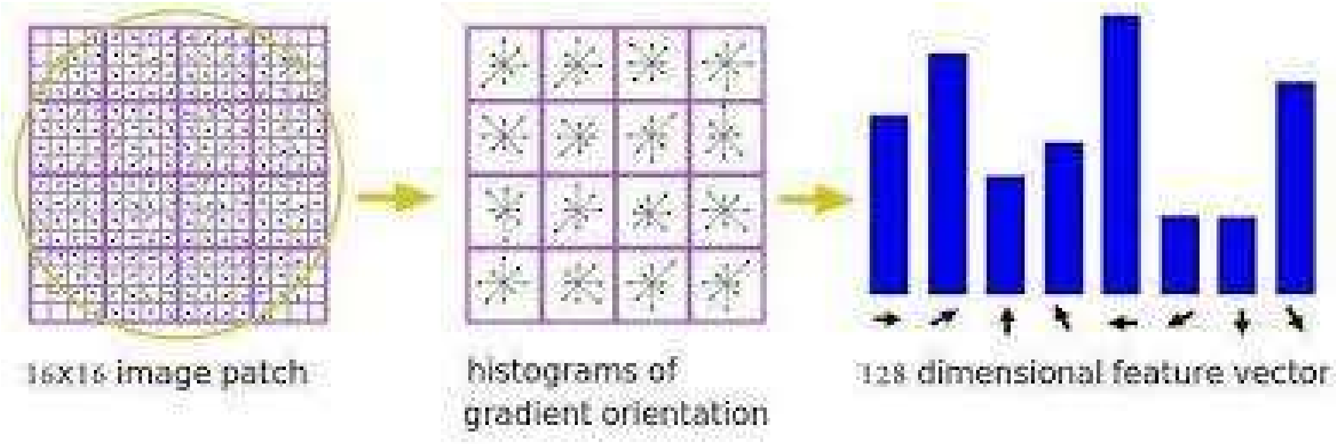
3. Descriptor matching
   – find best match from interest points in tracking model (Euclidean)
   - use hierarchical search structure (k-d tree / spill forest) for large descriptors
   - Approximate structure leads to incorrect matching (outliers)

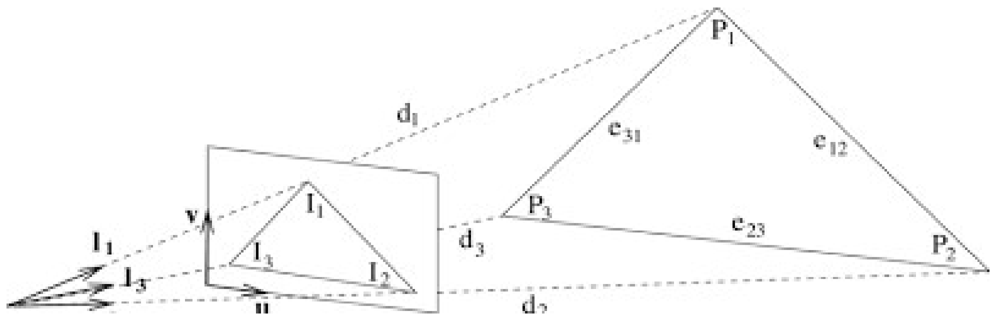
4. Perspective-n-Point (PnP) camera pose determination
   – recovering 6DOF pose
   - P3P algo gives 4 ambiguous solutions
   - n denotes no. of correspondences

5. Robust pose estimation
   - RANSAC, M- Estimator



SIFT descriptor – to built gradient orientation





Benefits of NFT:

- direct scene annotation
- pose stabilization
- extendible tracking range

P3P – computing distance from camera centre to 3D point

Incremental Tracking

Problem Addressed:

- Difficult to scaling the matching to large natural feature models
- Update rate in each consecutive frames drastically changes
- Coherence is ignored by detection method leading to wasting constrained computational resources

Solution: Incremental tracking

- Uses information from a previous step (recursive tracking)
- Track features across consecutive frames using search methods and combined with feature detection tracking

Components of Incremental Tracking:

- Incremental (active) search
- Interest point matching

Active search: initial estimate of camera pose is extrapolated from last known pose using motion model

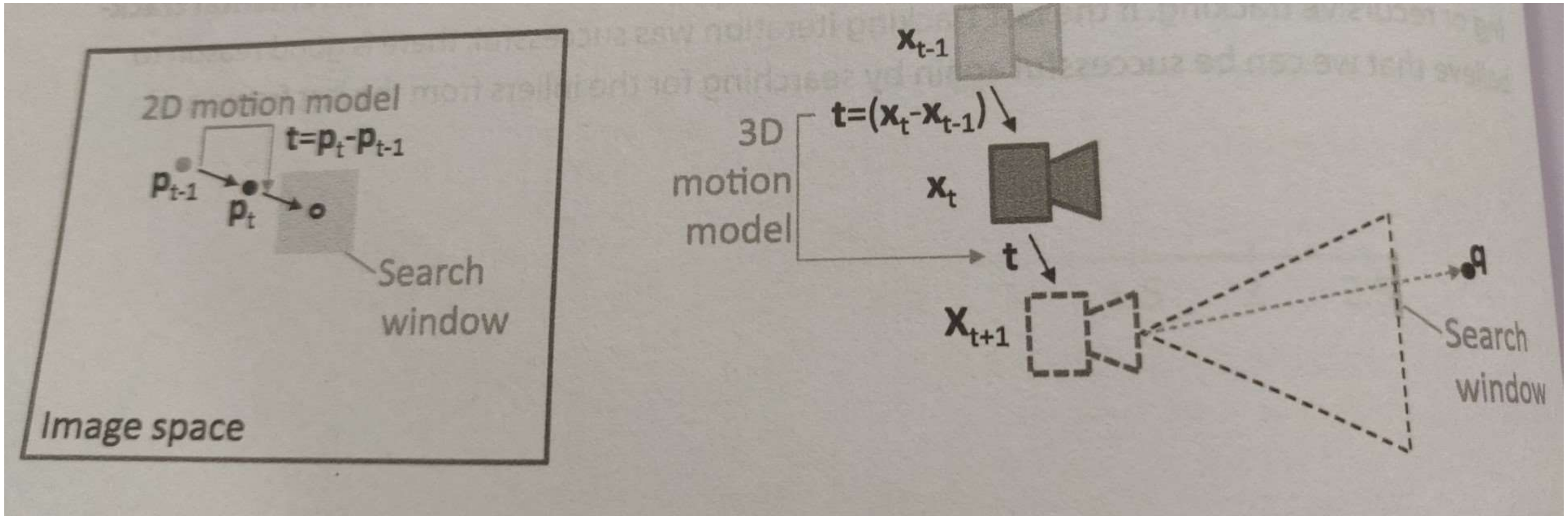
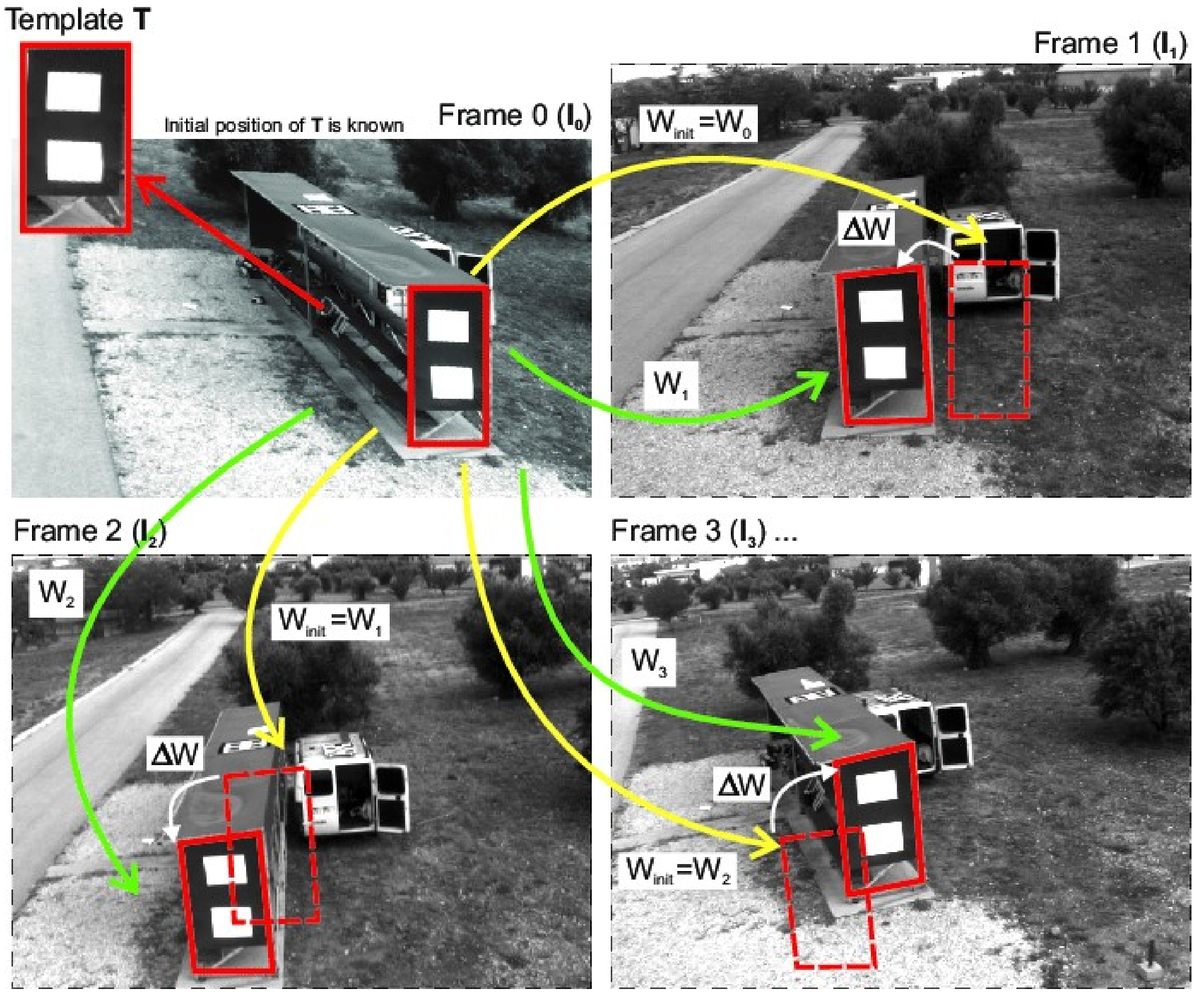


Figure 4.15 Active search in 2D (left). Active search in 3D (right).


Template **T**
Initial position of **T** is known
Frame 0 ($\mathbf{I}_0$)
Frame 1 ($\mathbf{I}_1$)
$W_{init} = W_0$
ΔW
W_1
Frame 2 ($\mathbf{I}_2$)
Frame 3 ($\mathbf{I}_3$) ...
W_2
$W_{init} = W_1$
ΔW
W_3
ΔW
$W_{init} = W_2$

Approaches:

- **Kanade-Lucas-Tomasi tracking (KLT)**
  - Extract points from initial image and track them using optical flow
  - Finds the parameters of warp that transforms a template image into input image
  - Minimising the intensity differences error

- **Zero-normalized cross correlation (ZNCC)**
  - Searching for *optimal position of a feature point* using optical flow
  - Only translation in the image plane is considered – disadvantage

- **Hierarchical search**
  - Determines the *final camera pose* in steps of decreasing magnitude
  - Resultant Camera pose is improved than pose from motion model

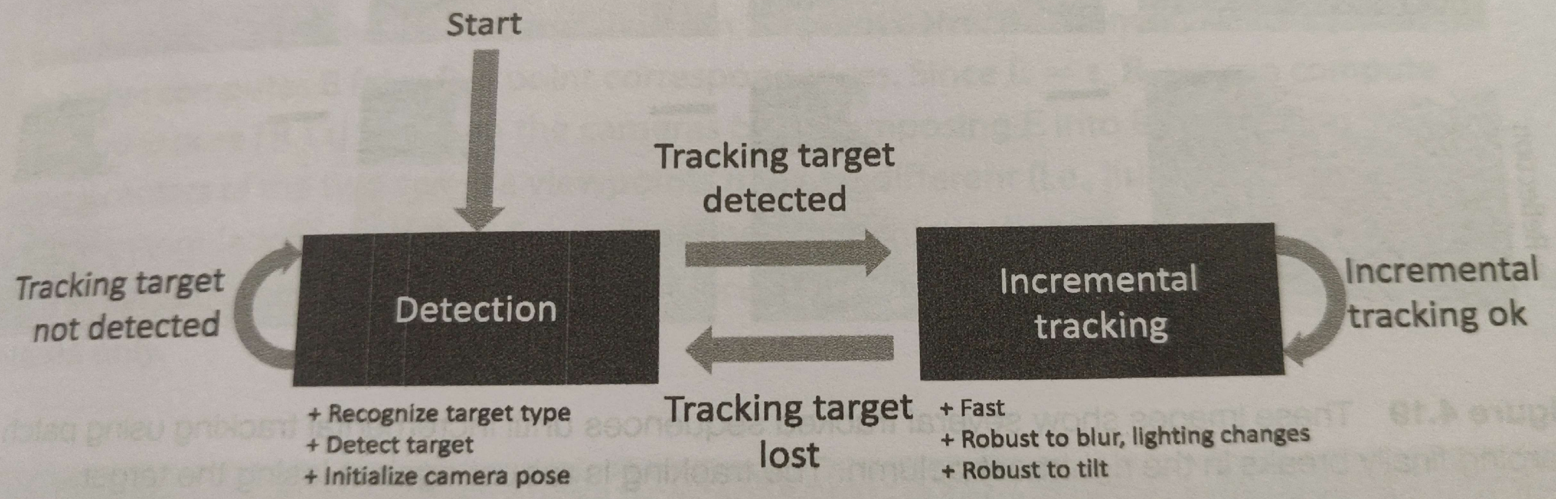
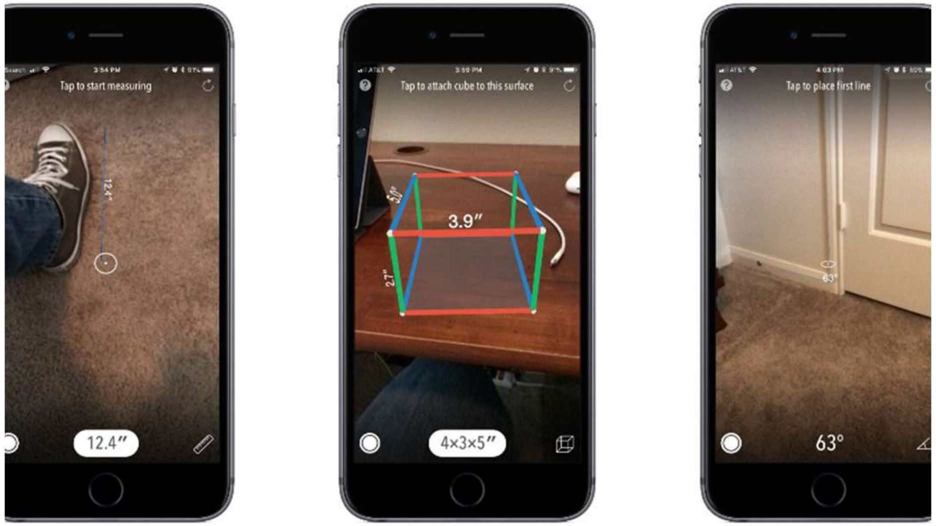
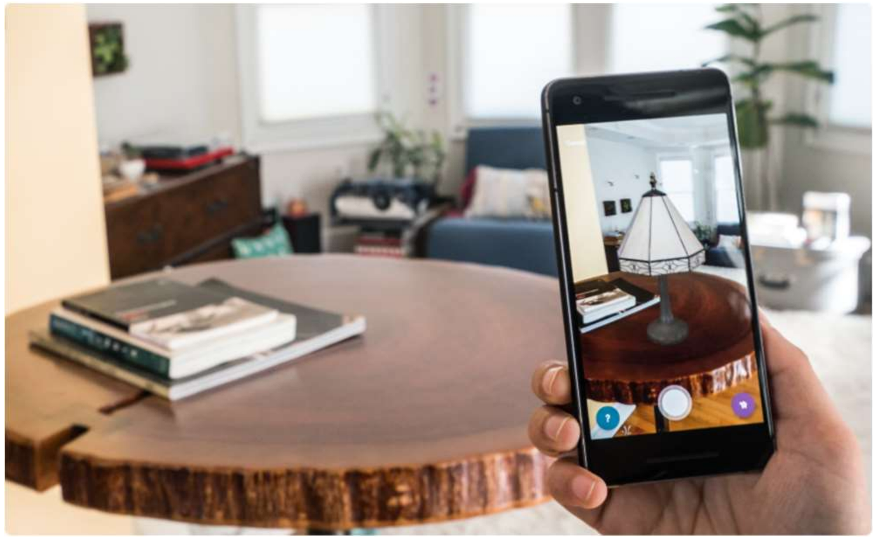


Figure 4.18 Tracking and detection are complementary approaches. After successful detection, the target is tracked incrementally. If the target is lost, the detection is activated again.

Incremental Tracking - Applications





AR Measuring Tool

The app measures the distance between two points using recognizable landmarks. Essential for measuring real-world spaces and placing virtual objects precisely.

AR Furniture Placement

The app recognizes the corners, edges, and surfaces of a room, allowing AR furniture to be placed in a precise location, with realistic scaling and positioning.

Simultaneous Localization And Mapping (SLAM)

- Explores pose computation, uses parallel/ dense tracking and mapping

1 How it works

A combination of mapping and tracking. The app builds a 3D map of the environment and tracks the device's movement in it. Essential for creating large-scale, immersive AR experiences.

2 Pros

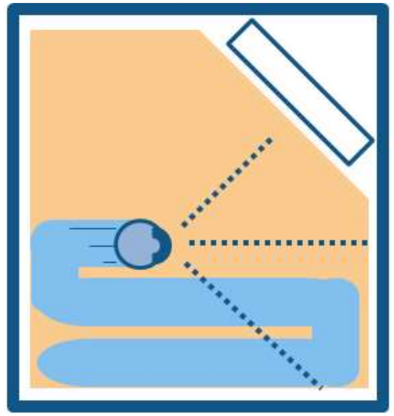
Allows for large-scale, immersive AR experiences, perfect for outdoor use, and complex indoor use cases.

3 Cons

Requires advanced processing power, could drain the battery, limited by the hardware of the device.

Examples - Simultaneous Localization and Mapping (SLAM)

Without SLAM:
Cleaning a room randomly.



With SLAM:
Cleaning while understanding the room's layout.

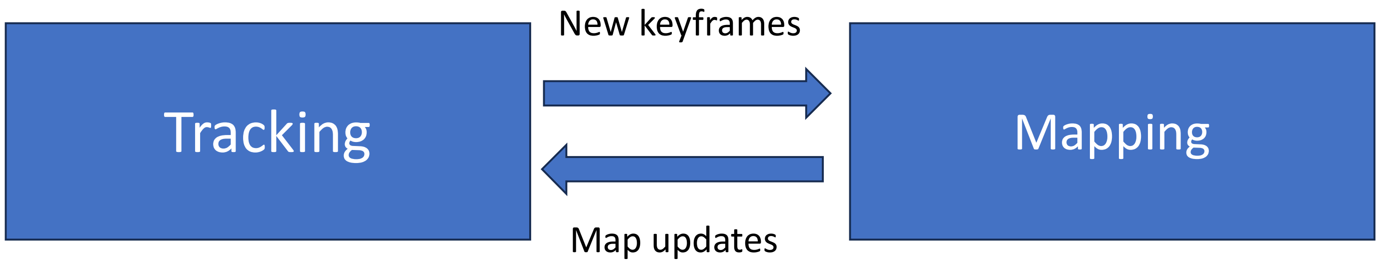
- Uses *sensor data to create a map of the environment* while simultaneously tracking object movement.
- Sometimes called as **visual odometry**

Pipeline steps:

1. Detect interest points in first frame
2. Track points from previous frame using KLT
3. Determine essential matrix between current and previous frames from feature correspondences (Five-Point Algorithm)
4. Recover incremental camera pose from essential matrix
5. Estimate scale through *Structure from motion approach*
6. Proceed to next frame

Challenges with SLAM

1. Localization errors accumulate, causing substantial deviation from actual values
2. Localization fails and the position on the map is lost
3. High computational cost for image processing, point cloud processing, and optimization



Estimate camera pose
for every frame

- Extend map
- Improve map
- slow update rate

Outdoor Tracking

- Tracking methods in wide area outdoor environments
- Requires scalable feature matching and assistance from sensor fusion and geometric priors

1 How it works

Uses GPS and other sensors to track the device's location and orientation. Ideal for creating AR experiences in large outdoor spaces.

2 Pros

Accessible, uses standard hardware, ideal for outdoor use.

3 Cons

May not be accurate, difficult to use in dense urban areas, relies on good weather and clear visibility.

Conclusion

The Future of AR

AR and computer vision technologies are rapidly advancing. With the help of machine learning, AR apps will be able to recognize and track more objects more accurately than ever before.

New Opportunities

The range of applications for AR is vast. From gaming and entertainment to education and industry, AR and computer vision technologies open up new opportunities for innovation and creativity.